

# **Common Home Reference Manual**

# Table of Contents

Preface .....	3
Introduction .....	4
Background .....	4
Advantages .....	5
Common Home Naming Scheme .....	6
Home Pathname .....	6
Host, Host-Group, and System Names .....	7
Example Home Directory .....	8
Home-Directory Features Explained .....	9
Master Dot Files .....	9
Host-Specific Dot Files .....	9
Personal Home Subdirectories .....	10
Backup Subdirectories .....	11
Quotas .....	12
Background .....	12
Reporting and Managing Quotas .....	13
GLOBAL_QUOTA (Obsolete) .....	15
Writing to Other Homes .....	15
Side Effects and Warnings .....	16
Home References .....	16
Mail .....	16
Checkpointing Problems .....	16
Home Permissions .....	16
Staff-Only Features .....	18
Migration From /u To /g .....	19
Common (Global) Code Directories .....	20
Directory Structure and Roles .....	20
Transition Management .....	23
Disclaimer .....	24
Keyword Index .....	25
Alphabetical List of Keywords .....	27
Date and Revisions .....	28

# Preface

- Scope:** The Common Home Reference Manual explains the purposes and features of, the naming scheme for, and the side effects of using common (globally mounted, host-shared) home directories on the LC computers. Included are the role of the HOST\_GRP and SYS\_TYPE environment variables in managing common-home files. The contents and organization of a typical common home directory are also shown, and the most common pitfalls are noted. This manual also explains the rules for common-home space quotas and the tools that monitor those quotas. Finally, LC also provides a common (globally mounted, host-shared) /usr/gapps file system for managing shared user application codes in the same way that common home directories manage individual files, and this manual explains how /usr/gapps is organized.
- Availability:** The introduction (just below) lists the machines that now support common home directories at LC.
- Consultant:** For help contact the LC customer service and support hotline at 925-422-4531 (open e-mail: lc-hotline@llnl.gov; SCF e-mail: lc-hotline@pop.llnl.gov).
- Printing:** The print file for this document can be found at:

OCF: <http://www.llnl.gov/LCdocs/chome/chome.pdf>  
SCF: [https://lc.llnl.gov/LCdocs/chome/chome\\_scf.pdf](https://lc.llnl.gov/LCdocs/chome/chome_scf.pdf)

# Introduction

## Background

### ROLE.

The home directories on these LC unclassified (OCF) hosts

- ASC IBM/AIX machines (UP),
- Linux (CHAOS) machines (Yana, Thunder, etc.),
- LUCY (and many other, special-purpose Suns),

and on these LC secure (SCF) hosts

- ASC IBM/AIX machines (UM, UV, Purple),
- Linux (CHAOS) clusters (Lilac, ACE, etc.),

reside on dedicated, NFS-mounted file systems ("network attached storage") that provide their own (open and secure) common (shared) home directory structure, in which the *same* home directory serves several different machines at once.

This common home directory offers a central location for files to be shared across multiple machines, eliminating the need for FTPing between these hosts or maintaining redundant sets of files. Your common home directory also provides a shared location for your start-up (.login or .profile) and run-control (other dot) files, and a consistent default arrival directory when you log onto any of these hosts.

For an overt comparative table that summarizes the intended use of all the major file systems on LC production machines (including the common home directories), with links to the technical features of each file system, consult the "File-System Usage Comparison" section of the EZFILES (URL: <http://www.llnl.gov/LCdocs/ezfiles>) basic guide.

### TEST.

If you have a single version of an application code (or utility) that you run on diverse machines, you may need to test for the presence or absence of LC common home directories to decide about where to locate (initialization or output) code-related files. To facilitate this test, each LC machine offers a system file called /etc/home.config, in which one line contains the keyword "FILE\_SYS" (uppercase) followed by one of four configuration values (shown below, mixed case). Your code can reveal the content of this line, for use in conditional tests, by executing

```
grep FILE_SYS /etc/home.config
```

The four possible configuration values that this might return are:

- |         |  |
|---------|--|
| global  | means that all LC global file systems, including the common home directories of <i>all</i> users, are available.                             |
| LC_only | means that only LC staff common home directories are available; home directories for general users are <i>not</i> mounted here (e.g., LUCY). |

green	means that only unrestricted-network (local) home directories are available (e.g., for "foreign national" users).
pl1	means that this SCF system at security level PL1 has <i>no</i> global (including common home) file systems available.

## Advantages

- You have only one home directory space to manage across several machines. You no longer need to keep redundant sets of files on multiple machines, or FTP (home) files between machines to keep them synchronized or to share new versions.
- You can make login customizations in one place and have them apply for all hosts, as well as still make host-specific login customizations.
- Common-home (/g) files are all automatically backed up (page 11) four times over the most recent 48 hours, and you can easily recover any of these backup versions if you mistakenly delete or spoil a current home file.
- The files in your common home directory are never purged (but there is a quota (page 12) on total size that you must not exceed). Your home quota varies by your status (LC staff members have smaller quotas than other LC users) but not by your network (OCF and SCF quotas are the same now for most users). See the Quotas (page 12) section below for current details.

# Common Home Naming Scheme

## Home Pathname

All open-environment common home directories reside on the /g (global) file system, accessed by the same pathname on each host:

```
/g/ggroup/uname  
[example: /g/g16/smith]
```

where

*ggroup* is an administrative subdirectory different from the UNIX groups that LS or GROUPS reports. On OCF machines, *ggroup* is g0 for LC staff and g2 or above for other users. On SCF machines, *ggroup* is g5 for LC staff and g10 or above for other users.

*uname* is your alphabetic login account name (not the numeral *uid* by which your file and block quotas are reported).

At this time, /g is on highly reliable RAID disks that are backed up (page 11) twice each work day and are located on multiple NFS (Network File Service) servers ("network attached storage").

On SCF machines, one secure common-home directory system (using children of /g, as shown above) is shared among the nodes of every ASC IBM/AIX machine and every Linux (CHAOS) node (IBM or other vendor) as well. Users whose SCF accounts were first created after October 1, 1999, automatically use a child of /g as their common home directory.

A similar but separate common-home directory system (using children of /u instead of /g) existed among only the nodes of the former Forest cluster of Compaq computers until December, 2001. At that time all /u users were automatically migrated to the larger and more comprehensive /g SCF common-home directory system, using file-migration procedures summarized in a separate section (page 19) below.

**WARNING:** When LC adds new file-system servers or retires old ones (usually to increase total capacity), the number of users allowed per file system may change, forcing some users to change their *ggroup* (e.g., from g16 to g10). To avoid negative side effects of such occasional absolute home pathname changes, LC urges you to use ~ or \$HOME instead whenever you refer to your common home directory.

## Host, Host-Group, and System Names

### HOST\_GRP.

With one common home directory serving many hosts that may have started with different control files or may need distinct run-control specifications, host names are often used to identify transitional subdirectories (such as /lucy) or machine-specific dot files (such as .login.lucy). But generally, machines with the same hardware are "alike" *unless they run Linux*. So to simplify such file naming for machines that are "alike," we have assigned like machines to a "host group" with a single designated host-group name. The current host groups are:

HOSTS:	SHARED HOST-GROUP NAME:
All IBM AIX clusters	
open (uPurple)	blue
secure (UM, etc.)	white
All Linux clusters (includes IBM BG/L)	linux
All LC Suns	sol_desktop

### SYS\_TYPE.

If you wish to create a file that applies to all machines with the same (brand or version of) *operating system*, you can use the same approach as above with these designated "system types":

NAME:	APPLIES TO:
SunOS_5	SunOS version 5.x (Solaris)
aix_5[_nn_xxx]	AIX version(s) 5.x (IBM UNIX), where nn and xxx are bit and switch details
chaos_3_nn_xxx	Linux CHAOS version 3 (on Intel hardware, where nn and xxx are chip and switch details (absent if no switch)

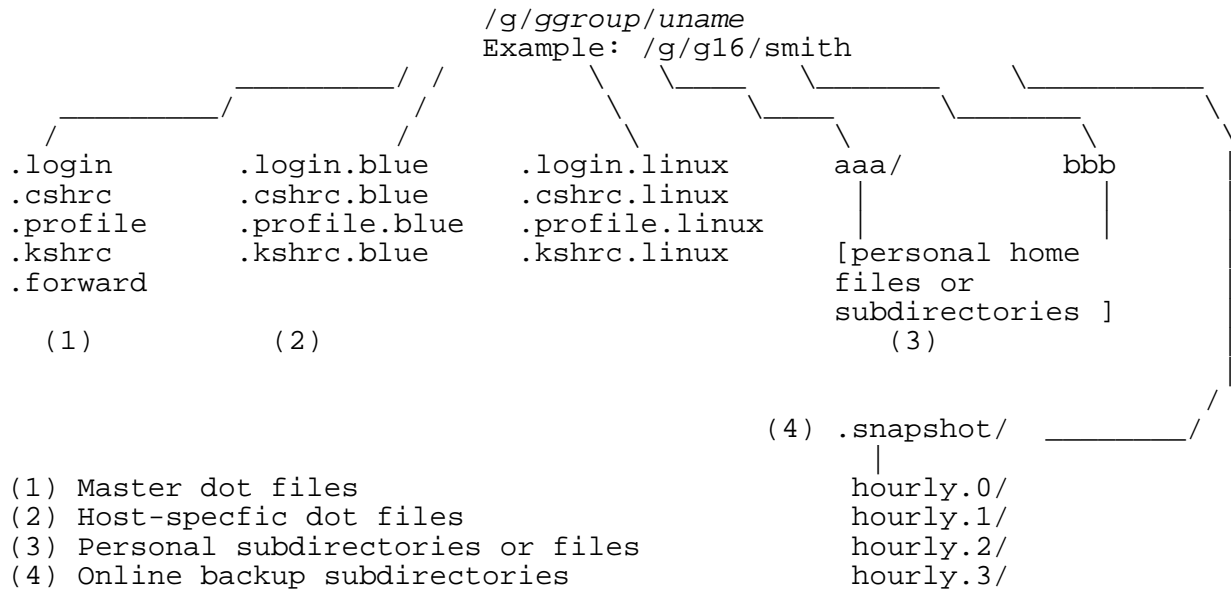
For example, to define aliases on all Solaris systems a ~/.cshrc.SunOS\_5 file can be created.

### ENVIRONMENT VARIABLES.

On each machine, the environment variable HOST\_GRP contains that machine's designated host-group name (e.g., on THUNDER that variable contains the string linux). Likewise, on each machine the environment variable SYS\_TYPE contains the designated system or platform type for that machine (e.g., on THUNDER that variable contains the string chaos\_3\_ia64\_elan4). See the "Get-Only Variables" [section](http://www.llnl.gov/LCdocs/ev/index.jsp?show=s3.2) (URL: <http://www.llnl.gov/LCdocs/ev/index.jsp?show=s3.2>) of LC's Environment Variables manual for other environment variables that LC systems use in this way to better manage jobs.

## Example Home Directory

For user smith with global subdirectory /g/g16 and with accounts on the open IBM/AIX and Linux/CHAOS clusters (the normal situation for OCF users), here is what the top level of the common home directory contains:



See the next section for an explanatory analysis of the contents of this directory, keyed to this example.



# Home-Directory Features Explained

## Master Dot Files

Each user's common home directory contains a set of master dot files for login and run-control support. These are called `.login`, `.cshrc`, `.profile`, and `.kshrc`, as shown at (1) along the left in the [example](#) (page 8) home directory above.

These master dot files:

- are copies of files with the same name in the directory `/gadmin/etc/skel`. If you ever inadvertently damage or destroy your version, you can restore it from that set.
- detect the current value of the environment variable `HOST_GRP` and then invoke ("source") the corresponding host-specific dot file (such as `.cshrc.linux`), discussed more in the next section.
- detect the current value of the environment variable `SYS_TYPE` and then invoke ("source") the corresponding system-specific dot file (if any, such as `.login.chaos_3_ia64_elan4`), discussed more in the next section.
- are the place to install your own customizations IF they are intended to apply to all hosts and all operating systems (for example, aliases that you want to work on any common-home machine). Host-specific customizations go into your host-specific dot files (below) instead.

These master dot files make it possible to use a single common home directory for logins on multiple hosts. Do not overwrite these files with host-specific versions of login files and do not add host-specific customizations to them.

## Host-Specific Dot Files

Host-specific dot files supplement your master dot files (described above). Each one has the relevant host name appended (such as `.login.lucy`) or the relevant `HOST_GRP` name if the file applies to all similar machines (such as `.login.compaq`). These host-specific dot files show at (2) in the preceding [example](#) (page 8). (You may have inherited some files like this from your old home directory before it was shared among several machines.)

These host-specific files are the place to install your own customizations that apply only to individual hosts (an alias to run one-host software, for example). You can replace these files, or modify them with whatever host-specific features you need, at any time. Each host-specific dot file is invoked ("sourced") by the corresponding master dot file when you log in.

## Personal Home Subdirectories

Most of the files in your common home directory are personal project files (such as bbb) or personal subdirectories (such as aaa/), as shown at (3) in the preceding [example](#) (page 8). Organizing most of your personal home files by subdirectory within your home directory can assist file management and file sharing as your collection of files grows large.

Users who had different home directories on different machines before LC installed common home directories have automatically inherited one personal subdirectory for each such machine (for example, one for the now-defunct CORAL and EAST machines). Preserved within these inherited transitional subdirectories are all the directories and files (with their old permissions and tree structure) formerly unique to their separate home directories. So you should look in a subdirectory called `/g/ggroup/uname/east` for old home files formerly in your separate EAST home directory, for example.

## Backup Subdirectories

Each user's common home directory automatically contains a special backup subdirectory called `.snapshot`, as shown at (4) in the preceding example (page 8). This `.snapshot` subdirectory is unusual because it is:

- Hidden. It is NOT reported by running `LS`, as the other dot-named children of your home directory are. But you can change directories (`CD`) into it to list and copy its files.
- Free. The disk-space quota on your common home directory (next section) does NOT apply to the backup file copies within `.snapshot`.

### BACKUP SCHEDULE:

The system automatically makes a complete backup copy of every file in your common home directory twice each day, timed to roughly cover half of each normal work day:

Noon  
7 p.m.

The four most recent backup copies reside in four subdirectories of your `.snapshot` directory in reverse chronological order as shown in this chart:

<code>.snapshot</code> Children -----	Backup Copy ----	Example for 3 p.m. Tues -----
<code>hourly.0</code>	most recent	Noon Tues
<code>hourly.1</code>	half day older	7 p.m. Mon
<code>hourly.2</code>	full day older	Noon Mon
<code>hourly.3</code>	3 half days older	7 p.m. Sun

### BACKUP PROPERTIES:

Each night LC makes an (incremental tape) backup copy of every common home directory for safety. But there is no easy way for individual users to access those system backups, which are really not intended for personal use. The four backup copies in the four `hourly.n` subdirectories of your `.snapshot` directory, however, are always on disk (online) and readily available directly to you at any time.

Whatever their apparent permissions (as originally set by you with `CHMOD` and as reported by the `-l` option of `LS`), all of your backup files in `.snapshot` are really read-only. Altering or editing them inside `.snapshot` is not allowed. You should copy (`CP`) any backup file(s) you need to somewhere else in your home directory, or to any work directory you choose (such as `/usr/tmp`), before you try to edit or otherwise reuse them.

# Quotas

## Background

Starting in October, 1997, all LC common home directories have quotas on the total size of files allowed (but not on the total number of files allowed). The quotas are large, but ignoring them can cause severe and immediate loss of data. The current common-home disk-space quotas by platform are:

Machine type	Home quota
-----	-----
Open IBMs, Compaqs	16 Gbyte (/g)[*]
Open Linux machines	16 Gbyte (/g)[*]
SCF IBMs, Compaqs	16 Gbyte (/g)[*]
SCF Linux machines	16 Gbyte (/g)[*]

[\*]For LC staff members only, home quotas are 8 Gbyte

On all LC production machines, an altered version of the UNIX QUOTA utility has been installed to report on your current home-directory disk usage and on your allowed quota. There is also a login warning that alerts you if you reach 90% of your common-home quota. A section below (page 13) explains the details, which unfortunately differ slightly among brands of machine. (The former reporting tool GLOBAL\_QUOTA is no longer needed, or available, anywhere. The LC customized tool BDF reports on total and used space for any specified file system, although not on your personal quota.)

To avoid hitting your common-home quota needlessly, LC suggests that you use /nfs/tmp\* or /usr/tmp instead of your home directory for big files that you personally archive, for files that do not need to be shared among machines, or for files that do not need the automatic backup that files in the common home directory receive. See EZFILES (URL: <http://www.llnl.gov/LCdocs/ezfiles>) for more related file-management suggestions to avoid wasting your common-home file quota.

# Reporting and Managing Quotas

## ROLE:

On all LC machines (open and secure) that share common home directories, you use (an altered version of) the standard UNIX QUOTA utility to report your current disk usage and your home quota. Quotas are reported by file-system name, so you need to look for the name of your home directory's file system (such as /g/g17 or /g/g0) in QUOTA's report. Usage is now reported in megabytes (M) or gigabytes (G) on all LC production machines. [NOTE that the former GLOBAL\_QUOTA reporting tool is no longer needed and has been replaced by a pointer to QUOTA on all machines.]

## EXECUTE LINE:

To run QUOTA to report on your common home directory (as well as on all other mounted global file systems, such as /nfs/tmp\*), type

```
quota -v
```

The program generates a usage report (only if you use -v; the default report is null) and then automatically ends.

## WARNING:

QUOTA *cannot* report data for /nfs/tmp1 or /nfs/tmp0 (the current global file systems other than the common home directories). LC is pursuing this problem. Local tool BDF does nicely report total and used space for *any* file system, but it cannot report on your personal quota of that space.

## OUTPUT:

On all LC production machines (including Linux/CHAOS machines), QUOTA reports usage in megabytes (M) or gigabytes (G), reports only on globally mounted file systems such as the common-home directories with -v, and uses these descriptive headers:

```
Disk quotas for jmw:
Filesystem  used   quota  limit  timeleft  files  quota  limit  timeleft
/g/g17      721.7M 16.0G   16.0G           2874   n/a    n/a
...
```

where

Filesystem	identifies which specific subdirectory (such as /g17) has its usage reported on this line. You must search for the one containing your home directory in a long numerical list of globally mounted file systems.
used	is your current total home-directory disk usage. Usage is reported in megabytes (M) or gigabytes (G) on all LC production machines now. Your <u>backup</u> (page 11) copies of home files in .snapshot are excluded from this total.
quota	is your quota threshold for the specified (sub)directory in gigabytes, where there is no "soft" quota or grace period before this threshold is enforced.
limit	is now identical with quota on all machines for all LC users (it was once distinct on some SCF machines).

timeleft	(or grace) is not reported at LC at this time.
files	is your current total number of files and directories (inodes) within the specified globally mounted directory. There is NO quota on the number of home files you can have (as the next two columns suggest). Your <u>backup</u> (page 11) copies of home files in .snapshot are excluded from this total.

#### QUOTA MANAGEMENT:

Running QUOTA with the -v option will warn you if you have used more than 90% of your open home-directory disk quota. Also, a check run automatically when you log in to any machine sharing common home directories will warn you if you have exceeded 90% of your quota at login. Remember, however, that there is no grace period for quota enforcement on LC machines.

If you exceed your common home disk quota, then you risk immediate and severe loss of data:

- All program output sent to a common home directory that is over quota is truncated.
- Any file moved to an over-quota home directory is truncated.
- Just opening any common-home file for editing while you are over quota will cause the loss of the ENTIRE file contents.

These are unfortunate results of the UNIX implementation of quotas that only your caution can mitigate. You must destroy (RM) files after exceeding your quota (and store your files archivally before the problem arises) to avoid any such inadvertent loss of data.

## **GLOBAL\_QUOTA (Obsolete)**

The former GLOBAL\_QUOTA common-home reporting tool has been replaced by (altered versions of) the UNIX QUOTA utility (page 13) on all LC machines.

## **Writing to Other Homes**

Most users write files into their own common-home directory only. However, your professional collaborations might sometimes require that you write files into the common-home directory of another user, or that you allow other users to write files into your common-home directory. Regardless of whether these different common-home directories happen to reside on the same (physical) file system (e.g., g18) or different file systems (e.g., g18 and g20), simply changing UNIX permissions with CHMOD (to allow group members to write to the target directory) will enable the cross-directory writing that you want. Special arrangements for multiple quotas, required before September, 1998, are no longer needed on LC open machines. If no existing group covers you and your desired collaborators, just contact the LC Hotline and request creation of a new group.

Because of the danger of lost data or hidden changes to your login files, WORLD write access to your top-level common home directory (i.e., to /g/ggroup/uname) is NOT allowed on open machines. And world or even group write access to /g/group/uname disables SSH as well. You can enable world or group write access to your home subdirectories if you wish, but this too is risky.

Several alternatives to world access for file sharing are available on LC production machines (GIVE, TAKE, group access in nonhome directories, group access in storage). Check the section called "File-Sharing Alternatives Compared" in the EZFILES (URL: <http://www.llnl.gov/LCdocs/ezfiles>) basic guide for an overt explanation of the strengths and weaknesses of each alternative, with links to the technical usage details elsewhere in EZFILES.

# Side Effects and Warnings

## Home References

The most robust approach to referencing your common home directory on any machine is to avoid hard pathnames and instead use:

```
Your home directory:      ~    or    $HOME
xyz's home directory:    ~xyz
```

## Mail

Even while sharing one common home directory, each host receives its mail independently. And each host-specific .forward file still works, directing mail received by the host in its name (so .forward.east controls mail sent to east.llnl.gov).

You can create a master .forward file to direct all incoming mail for all hosts that share the common home directory if you prefer. But if your LLNL postmaster address user@llnl.gov points to one of the hosts covered, then do NOT use user@llnl.gov as the forwarding address in the universal .forward file (your mail would loop endlessly). You CAN safely use as a forwarding address in the universal .forward file the name of any single host (e.g., user@thunder.llnl.gov), even though the common home directory applies to that host. (All .forward files must remain world readable to function correctly.)

## Checkpointing Problems

Checkpointing a batch job allows you to restart the job after system problems and preserve any partial results generated before the last checkpoint. However, you can not checkpoint a job any of whose files reside on NFS-mounted disks. Since the current implementation of all LC common home directories relies on NFS-mounted disks, any batch job running on a common-home machine that spawns a shell will access its dot files on those disks, and hence the job will no longer checkpoint.

## Home Permissions

Your common home directory is created owned by you as user and as group. These default permissions allow access only to you, the owner. If you previously allowed access by other users to your home directory on a specific host, then you will need to decide if you wish to allow that same access to your common, host-shared home directory (remember that it would then be open to many more users than just those of the original host) and modify the permissions with CHMOD.

Because of the danger of lost data or hidden changes to your login files, WORLD write access to your top-level common home directory (i.e., to /g/ggroup/uname) is NOT allowed on open LC machines. And world or even group write access to /g/group/uname disables SSH as well. You can enable world or group write access to your home subdirectories if you wish, but this too is risky. Consider using GIVE and TAKE to share files instead.

Several alternatives to world access for file sharing are available on LC production machines (GIVE, TAKE, group access in nonhome directories, group access in storage). Check the section called "File-Sharing



Alternatives Compared" in the EZFILES (URL: <http://www.llnl.gov/LCdocs/ezfiles>) basic guide for an overt explanation of the strengths and weaknesses of each alternative, with links to the technical usage details elsewhere in EZFILES.

## Staff-Only Features

These features of the common home directory affect primarily or exclusively LC staff members, not general users:

**Xauth**                      Xauth can be used as a better alternative to xhost for X-terminal display authorization.

**MacX**                      Master dot files support MacX by automatically re-executing your .login file after you log in.

**Terminal support**

X-terminal users may need to move (or merge) their .Xdefaults files and .ssh users will need to set up a ~/.ssh directory (copy the identity.pub file to the authorized\_keys file).

**FTP**                      When you use FTP to connect to your desktop workstation you will arrive in your root directory (/), not your home directory. You can either use CD to move to your home directory during the FTP session, or (to reach your common-home files) you can FTP to some other host that shares the common home directory.

## Migration From /u To /g

By December 6, 2001, all SCF users had been migrated from the former SCF /u (local) home directory systems to the global SCF /g home directories. Many users moved some or all of their home files themselves, to their own desired locations under /g. For those who did nothing, the LC system administrators performed the following "default migration" of the former /u files (no files were simply deleted).

### ALL /U FILES EXCEPT DOT FILES:

- Users with no previous /g home directory had all of their (nondot) /u files moved to the TOP level of their new /g home directory. For example, if user joe had no /g home directory before this forced migration, then all files in /u/u10/joe were migrated to /g/g10/joe (not all users have the same /g numbered subdirectory as their old /u numbered subdirectory).
- Users who already had a /g home directory (for example, on any SCF IBM machine) had all of their (nondot) /u files moved into a subdirectory of their /g home directory called FOREST (named for a now-departed computing cluster). For example, if user jill had a previous /g home directory, then all (nondot) files in /u/u10/jill were migrated to /g/g10/jill/forest.

### FORMER /U DOT FILES:

- Forest-specific dot files were copied from the /u home directory to the TOP level of the /g home directory. For example, /u/u10/joe/.cshrc.forest was copied to /g/g10/joe/.cshrc.forest.
- MASTER dot files were copied from the /u home directory to the forest subdirectory of the /g home directory (regardless of the user's previous /g status). For example, /u/u10/joe/.cshrc was copied to /g/g10/joe/forest/.cshrc. From here, each user can copy, move, or reuse all or part of each master dot file depending on their individual needs.
- All other /u dot files (if any) were copied into the FOREST /g subdirectory AND, if no file of the same name already existed in the top level of the /g home directory, they were copied into the top level of the /g home directory as well (so every user has a copy in FOREST and some have a second copy in the top level /g home).
- All .Xauthority files were merged in the /g home directory.

As a result of these default migrations, SCF users must replace any hard-coded references to their former /u directories in pathnames, scripts, aliases, or variables with either a corresponding reference to /g (based on the above locations) or a soft-coded reference to \$HOME. Note also that any special quota extensions granted to a former /u home directory were maintained for the corresponding /g home directory. To avoid file-name collisions, these /forest subdirectories of /g persist (until you personally empty and remove them), even though the Forest cluster for which they were named itself disappeared from SCF in October, 2002.

# Common (Global) Code Directories

## Directory Structure and Roles

Beginning in May, 2001, an NFS-mounted global (common) file system called `/usr/gapps` appeared on all LC production machines (one `/usr/gapps` system is mounted on all SCF production machines, and a comparable `/usr/gapps` system is mounted on all OCF production machines).

Each `/usr/gapps` system behaves like the common (globally mounted) home directories, except that the intended use is to manage the shared user application codes formerly residing in each machine's local `/usr/apps` directory.

### ADVANTAGES.

Because `/usr/gapps` is globally accessible, it offers shared-code managers advantages similar to those that common home directories offer to individual users:

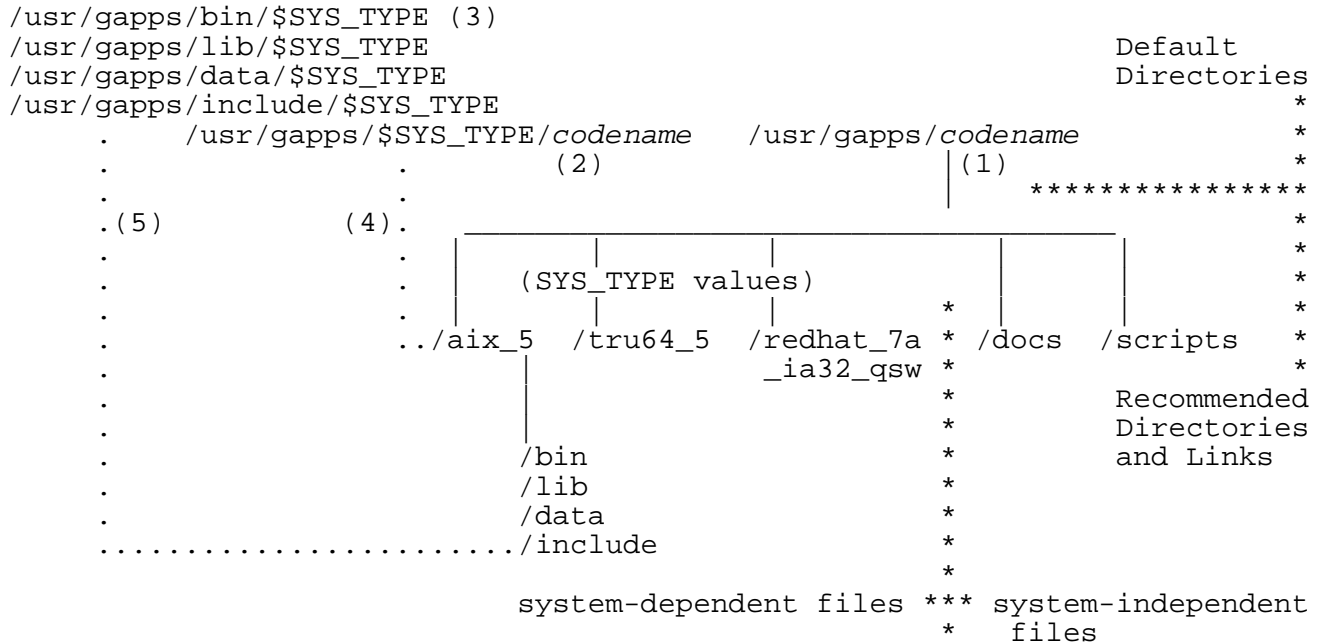
- Redundant, system-independent files and directories on each separate machine are no longer needed.
- Keeping system-independent files consistent across machines becomes automatic.
- Moving files between machines is unnecessary; all file management can occur on any single machine that shares `/usr/gapps`.
- As with home directories, 48 hours of automatic online backup copies are provided for all code files in `/usr/gapps` (and there is no size quota).

### DISADVANTAGES.

A more elaborate directory structure is needed for two reasons:

- With one global `/usr/gapps` file system (each on OCF and SCF), all system-*dependent* versions of a code's binary, data, library, and include files end up in the same directory tree. So separate subdirectories for each machine "type" (technically, for each possible value of the `SYS_TYPE` environment variable) are needed to keep system-dependent files for the same code properly segregated. (As a [previous section](#) (page 7) explains, a `SYS_TYPE` is basically a brand of operating system.)
- To maintain all the paths and environment variables that formerly led to codes in each local `/usr/apps`, a fairly elaborate set of shadow directories and links is needed as a front end for the actual files in `/usr/gapps`. The diagram below shows these extra path-preserving complexities, and the [next subsection](#) (page 23) explains how the transition from the old to the new file structure is handled.

## /usr/gapps And Its Children



In this structure, the default `/usr/gapps` subdirectories appear at the top of the diagram, while the recommended (but optional) child subdirectories appear below the `***` line. The directories have these roles (for each *codename*):

(1) `/usr/gapps/codename`

holds the actual files associated with *codename*, either in system-independent subdirectories (such as /docs, on the right side of the diagram), or in a tree of system-dependent subdirectories (left side, such as /aix\_5 and its children, one for each currently allowed value of SYS\_TYPE (page 7)).

(2) /usr/gapps/\$SYS\_TYPE/*codename*

holds only links to the actual files and subdirectories that are really in `/usr/gapps/codename` (above). Other files and directories created here will be periodically removed. This set of directories (one for each value of `SYS_TYPE` (page 7)) exists solely to support certain old paths and moves that cannot be maintained *automatically* in any other way. For example,

```
cd /usr/apps
cd codename
```

can automatically translate into

```
cd /usr/gapps/$SYS_TYPE
cd codename
```

but not into `cd /usr/gapps/codename/$SYS_TYPE`. So this helper-directory structure serves purely as a path-preserving front end to the actual structure of `/usr/gapps`.

(3) `/usr/gapps/bin/$SYS_TYPE`

(etc.) are default children of `/usr/gapps` that can either hold links (5) to system-dependent specific files (used during code execution) that actually reside in the `/usr/gapps/codename` tree, or optionally, hold the files themselves (if needed to avoid permission problems).

(4) is a recommended link between (2) and a child of (1) to support old paths and moves. See (2) above for an example of use.

(5) is a recommended link between (3) and a child of (1), enabling all actual files to reside down the `/usr/gapps/codename` tree rather than be scattered about in `/usr/gapps`.

## Transition Management

Code managers are free to organize their shared-code files as they see fit within the /usr/gapps structure. However, by combining the default directories shown in the diagram above (page 20) with a scheme of links (some default, some recommended for code managers to create when they are ready), most (perhaps all) formerly successful paths, environment variables, and moves among directories in /usr/apps can be preserved for the convenience of user scripts even after the switch to /usr/gapps. During the transition (May to August 14, 2001) but not before or after, LC will provide a temporary exact copy of the former /usr/apps called /usr/oldapps. Here are the links that most smoothly blend the old and new directories:

### LINKS DURING TRANSITION (MAY to AUGUST, 2001):

- Defaults:  
/usr/apps --> /usr/gapps/\$SYS\_TYPE  
/usr/gapps/\$SYS\_TYPE/*codename* --> /usr/oldapps/*codename*
- Recommended (to handle system-dependent files smoothly as soon as you have moved them to /usr/gapps and you no longer need /usr/oldapps):  
/usr/apps/*codename* --> /usr/gapps/*codename*/\$SYS\_TYPE  
/usr/gapps/\$SYS\_TYPE/*codename* --> /usr/gapps/*codename*/\$SYS\_TYPE

### LINKS AFTER TRANSITION (After AUGUST, 2001):

- Defaults:  
/usr/oldapps disappears completely, with all files destroyed.
- Recommended:  
/usr/apps/*codename* -->  
/usr/gapps/\$SYS\_TYPE/*codename* --> /usr/gapps/*codename*/\$SYS\_TYPE

# Disclaimer

---

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government thereof, and shall not be used for advertising or product endorsement purposes.

(C) Copyright 2007 The Regents of the University of California. All rights reserved.

---



# Keyword Index

To see an alphabetical list of keywords for this document, consult the next section (page 27).

Keyword	Description
<u>entire</u>	This entire document.
<u>title</u>	The name of this document.
<u>scope</u>	Topics covered in this document.
<u>availability</u>	Where this scheme applies.
<u>who</u>	Who to contact for assistance.
<u>introduction</u>	Overview of common home dirs.
<u>background</u>	Scope, goals of the project.
<u>advantages</u>	Common-home benefits listed.
<u>names</u>	Common home naming scheme.
<u>home-pathname</u>	Home pathname parts explained.
<u>host-groups</u>	Shared names for like hosts, systems.
<u>sys-types</u>	Shared names for like operating systems.
<u>example</u>	Typical common home dir, contents.
<u>home-features</u>	Common home contents analyzed.
<u>master-dot-files</u>	Top-level, global dot files.
<u>host-specific-files</u>	Dot files for one host, group, system.
<u>subdirectories</u>	Personal home subdirectories.
<u>backup</u>	Backup (.snapshot) home copies.
<u>quotas</u>	File size and number quotas on home.
<u>quota-background</u>	General role of home quotas, tools.
<u>quota-management</u>	How QUOTA helps manage quotas.
<u>global-quota</u>	GLOBAL_QUOTA now obsolete.
<u>other-homes</u>	Writing to someone else's home dir.
<u>side-effects</u>	Impact of, warnings about common home.
<u>home-references</u>	How to reference common home.
<u>mail</u>	Effects on .forward mail control.
<u>checkpointing</u>	Checkpoint thwarted by NFS.
<u>permissions</u>	Access issues for common home dir.
<u>staff-features</u>	LC staff-only home features.
<u>xauth</u>	Xauth support available.
<u>macx</u>	MacX log-on support available.
<u>xdefaults</u>	Changes to .Xdefaults needed.
<u>ssh</u>	Changes to .ssh needed.
<u>ftp</u>	Changed behavior of FTP.
<u>home-migration</u>	Default /u-to-/g moves summarized.

global-code-directories

Global /usr/gapps code directories.

gapps-structure

Content, organization of /usr/gapps.

gapps-migration

Transitional gapps defaults, tips.

index

The structural index of keywords.

a

The alphabetical index of keywords.

date

The latest changes to this document.

revisions

The complete revision history.

# Alphabetical List of Keywords

Keyword	Description
-----	-----
<u>a</u>	The alphabetical index of keywords.
<u>advantages</u>	Common-home benefits listed.
<u>availability</u>	Where this scheme applies.
<u>background</u>	Scope, goals of the project.
<u>backup</u>	Backup (.snapshot) home copies.
<u>checkpointing</u>	Checkpoint thwarted by NFS.
<u>date</u>	The latest changes to this document.
<u>entire</u>	This entire document.
<u>example</u>	Typical common home dir, contents.
<u>ftp</u>	Changed behavior of FTP.
<u>gapps-migration</u>	Transitional gapps defaults, tips.
<u>gapps-structure</u>	Content, organization of /usr/gapps.
<u>global-code-directories</u>	Global /usr/gapps code directories.
<u>global-quota</u>	GLOBAL_QUOTA now obsolete.
<u>home-features</u>	Common home contents analyzed.
<u>home-migration</u>	Default /u-to-/g moves summarized.
<u>home-pathname</u>	Home pathname parts explained.
<u>home-references</u>	How to reference common home.
<u>host-groups</u>	Shared names for like hosts, systems.
<u>host-specific-files</u>	Dot files for one host, group, system.
<u>index</u>	The structural index of keywords.
<u>introduction</u>	Overview of common home dirs.
<u>macx</u>	MacX log-on support available.
<u>mail</u>	Effects on .forward mail control.
<u>master-dot-files</u>	Top-level, global dot files.
<u>names</u>	Common home naming scheme.
<u>other-homes</u>	Writing to someone else's home dir.
<u>permissions</u>	Access issues for common home dir.
<u>quota-background</u>	General role of home quotas, tools.
<u>quota-management</u>	How QUOTA helps manage quotas.
<u>quotas</u>	File size and number quotas on home.
<u>revisions</u>	The complete revision history.
<u>scope</u>	Topics covered in this document.
<u>side-effects</u>	Impact of, warnings about common home.
<u>ssh</u>	Changes to .ssh needed.
<u>staff-features</u>	LC staff-only home features.
<u>subdirectories</u>	Personal home subdirectories.
<u>sys-types</u>	Shared names for like operating systems.
<u>title</u>	The name of this document.
<u>who</u>	Who to contact for assistance.
<u>xauth</u>	Xauth support available.
<u>xdefaults</u>	Changes to .Xdefaults needed.

## Date and Revisions

Revision Date -----	Keyword Affected -----	Description of Change -----
05Feb07	<u>background</u> <u>home-pathname</u> <u>host-groups</u> <u>example</u> <u>quotas</u>	Compaq machines, examples deleted. Compaq machines, examples deleted. Compaq HOST_GRP, SYS_TYPE deleted. BLUE replaces COMPAQ cases. Compaq references deleted. BDF tool reports used space.
23Jan06	<u>introduction</u> <u>sys-types</u>	Machine examples updated. CHAOS 3 becomes a SYS_TYPE. Cross ref added to Env. Vars. manual.
01Jun05	<u>host-groups</u>	HOST_GRP role, values updated.
07Mar05	<u>host-groups</u> <u>example</u>	WHITE, AIX roles clarified. LINUX replaces BLUE cases.
06Jul04	<u>sys-types</u> <u>master-dot-files</u>	CHAOS 2.0 becomes a system type. CHAOS 2.0 becomes a system type.
03May04	<u>quotas</u>	Details changed, /nfs/tmp SCF warning added.
03Nov03	<u>sys-types</u> <u>gapps-structure</u>	Linux, AIX system types updated. New system types replace old.
10Sep03	<u>home-pathname</u> <u>quotas</u>	/g/g5 for staff on SCF now. Bigger SCF quotas, standard QUOTA output.
04Jun03	<u>background</u>	How to test for FILE_SYS.
15Apr03	<u>introduction</u> <u>home-pathname</u> <u>quotas</u>	New OCF home quotas noted. Ggroups changed, home reference advice. New OCF home quotas noted.
07Jan03	<u>introduction</u> <u>sys-types</u> <u>index</u>	ILX cluster added. SYS_TYPE for ILX explained. Separate SYS_TYPE(s) keyword added.
07Oct02	<u>host-groups</u> <u>home-migration</u> <u>gapps-structure</u>	HOST_GRP, SYS_TYPE details updated. Forest name persists, cluster gone. Tru64_4 gone, Redhat added.
17Sep02	<u>introduction</u> <u>home-pathname</u> <u>other-homes</u> <u>checkpointing</u> <u>permissions</u>	SCF Linux clusters added. DFS reference deleted. DFS reference deleted. DFS reference deleted. DFS reference deleted.
06Mar02	<u>introduction</u> <u>names</u>	New clusters replace old. Compaq, Tru64_5 added.

	<u>example</u>	Details updated.
	<u>home-features</u>	Compass examples replaced.
	<u>quotas</u>	Compaq details updated.
03Jan02	<u>introduction</u>	Machine list updated.
	<u>names</u>	/u replaced by /g on SCF.
	<u>quotas</u>	/u information deleted.
	<u>home-migration</u>	Summarizes /u disposal on SCF.
	<u>index</u>	Revised migration description.
01May01	<u>global-code-directories</u>	
		New section explains new /usr/gapps.
	<u>index</u>	New keywords for new section.
25Sep00	<u>introduction</u>	Link to file-system comparison added.
	<u>quotas</u>	SCF home quotas increased.
	<u>other-homes</u>	Link to file-sharing tips added.
	<u>scope</u>	Revised printing instructions.
06Mar00	<u>entire</u>	All CRAY/J90 references deleted.
06Dec99	<u>introduction</u>	Role of /g expanded (SCF).
	<u>names</u>	/u and /g contrasted (SCF).
	<u>quotas</u>	/u and /g contrasted (SCF).
	<u>home-migration</u>	How to move from /u to /g (SCF).
09Jun99	<u>advantages</u>	Meiko (Tribble) role deleted.
23Mar99	<u>quotas</u>	New values, tools changed.
	<u>global-quota</u>	Now obsolete, use QUOTA.
	<u>home-pathname</u>	SCF names clarified.
	<u>example</u>	Sample home updated.
	<u>side-effects</u>	Obsolete sections removed.
	<u>permissions</u>	No WORLD access allowed.
	<u>index</u>	Obsolete keywords removed.
20Jan99	<u>master-dot-files</u>	
		/gadmin/etc replaces /g/etc.
	<u>subdirectories</u>	/gadmin/etc replaces /g/etc.
15Dec98	<u>other-homes</u>	World write access forbidden.
15Sep98	<u>backup</u>	New section on backup copies.
	<u>open-quotas</u>	New section on QUOTA role [since deleted].
	<u>other-homes</u>	Replaced multiple-quota section.
	<u>global-quota</u>	Now SCF use only.
	<u>introduction</u>	Tera cluster added.
	<u>example</u>	.snapshot diagram added.
	<u>quotas</u>	Open, SCF treatments diverge.
	<u>index</u>	Three new keywords added.
11Mar98	<u>multiple-quotas</u>	
		Diagnosis and advice added.
03Feb98	<u>quotas</u>	500-Mb Forest quotas noted.
	<u>introduction</u>	SCF Forest cluster added.
	<u>names</u>	SCF Forest cluster added.
	<u>who</u>	SCF e-mail added.
28Oct97	<u>quotas</u>	File quotas gone, DU note added.

01Oct97	<u>introduction</u> <u>host-groups</u> <u>quotas</u>	Removed retired machines. Removed retired machines. Updated GLOBAL_QUOTA run location, report content, current limits, and file-loss danger.
01Jul97	<u>background</u> <u>quotas</u>	Role of .profile noted. Use of /usr/tmp urged.
04Jun97	<u>background</u> <u>home-pathname</u> <u>host-groups</u> <u>quotas</u> <u>index</u>	Updated participating host list. Secure /u approach explained. Sky host added. Details, subsections added. Four more keywords added.
09Jan97	<u>background</u> <u>host-groups</u> <u>advantages</u> <u>names</u> <u>quotas</u> <u>staff-features</u> <u>index</u>	Blue host added. Blue host, AIX added. More benefits listed. User name (not uid) noted. New section added. New section added. Four new keywords added.
12Dec96	<u>entire</u>	First edition of COMMON.HOME manual.

TRG (05Feb07)

UCRL-WEB-200093

Privacy and Legal Notice (URL: <http://www.llnl.gov/disclaimer.html>)

TRG (05Feb07) Contact on the OCF: [lc-hotline@llnl.gov](mailto:lc-hotline@llnl.gov), on the SCF: [lc-hotline@pop.llnl.gov](mailto:lc-hotline@pop.llnl.gov)